DAQ Monitor Tutorial

T. Yasuda

Fermilab

Introduction

Purposes

- Performance monitoring during a run.
- Performance tuning tool during commissioning.
- Diagnostic tool in case of problems.

Program

- Server (DM_Server) and Client (DAQ_Monitor.py)
 system.
- Communicates with DAQ system elements using itc.
- Client displays information using Python/Tkinter GUI modules.

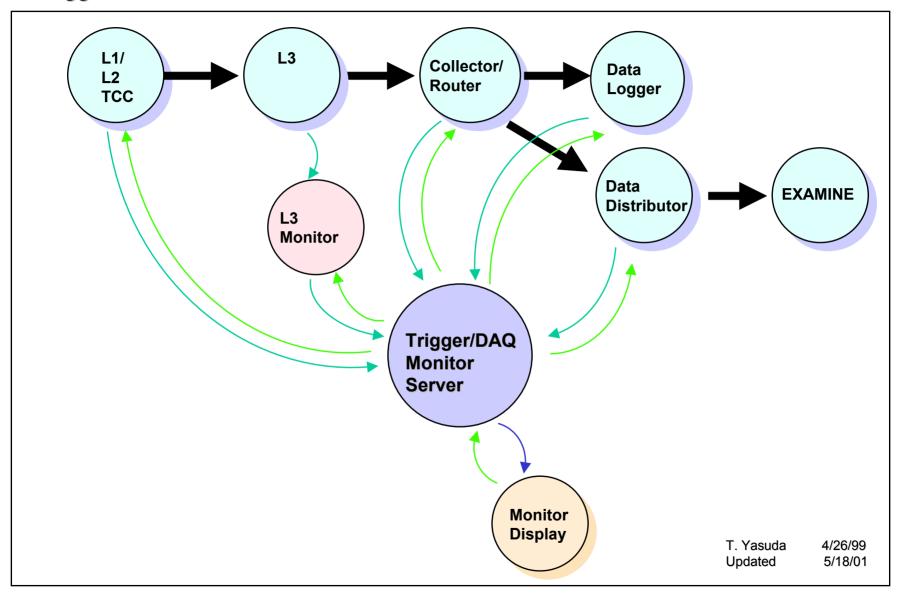
How it works

- DM_Server is:
 - an itc server with an itc processor to handle messages.
 - Contains a DM_Task class object.
 - DM_Task class object contains client objects for DAQ processes, e.g. TCC, Distributor, etc.
 - The clients objects are private data members of DM_Task class.
 - Data received from the DAQ processes as an itc message are handled by a callback method of the client object that requested the data.

How it works

- Requests data to the DAQ processes by sending an itc message.
- Communicates with receiver clients (Display clients).
- Receiver clients send a string message to request data from the DM_Server.

Trigger/DAQ Monitor



Software

- A cvs package: daq_monitor
- Written in C++ (server) and python (display client).
- Uses itc for communication.
- Different message types for individual sub-processes that run C++ server/client.
- String and opaque messages only for python (display) clients.
- Messages are encoded/decoded in XDR.
- Compiles on Linux.

Instructions

- To start the DAQ Monitor server
 - setup d0online
 - start daq monitor server
 - or start_daq daq_monitor_server
- To start a DAQ monitor display client (from any online node)
 - setup d0online
 - start_daq_monitor_display
 - or start_daq daq_monitor
 start_daq daq_monitor_small

